



Slice it Up

Procedural and non-destructive way
to slice spline objects in Cinema 4D

manual for v1.2

Slice it Up

with a click of a button

Installation

Unpack the archive and copy/paste entire **Slice it Up v1.2** folder to Cinema's **Plugins** folder:

- Windows: **Program Files\MAXON\Cinema 4D Rxx\plugins**
- Mac OS: **Applications/MAXON/Cinema 4D Rxx/plugins**

Folder structure should look like this:

- MAXON
 - Cinema 4D Rxx
 - plugins
 - Slice it Up v1.2
 - res (*folder*)
 - Slice it Up.pdf
 - Slice it Up.pype
 - Slice it Up.pypv

Once installation is finished run the plugin in Cinema 4D by clicking **Plugins > Slice it Up**. (R12 users will find plugin under **Python > Plugins > Slice it Up**).

How to Slice it Up

Drag and drop a closed spline to **Spline Object** field under **Generate** tab. Plugin will cut it into 10 equal width (by default) pieces and lay them side by side. In order to generate geometry, navigate to **Extrude** tab and enable **Generate Mesh** checkbox. Now it's render ready.

However, it does not look too appealing right away. So go ahead and change some values under **Generate** tab. Learn more about controls by reading this documentation.

Interface

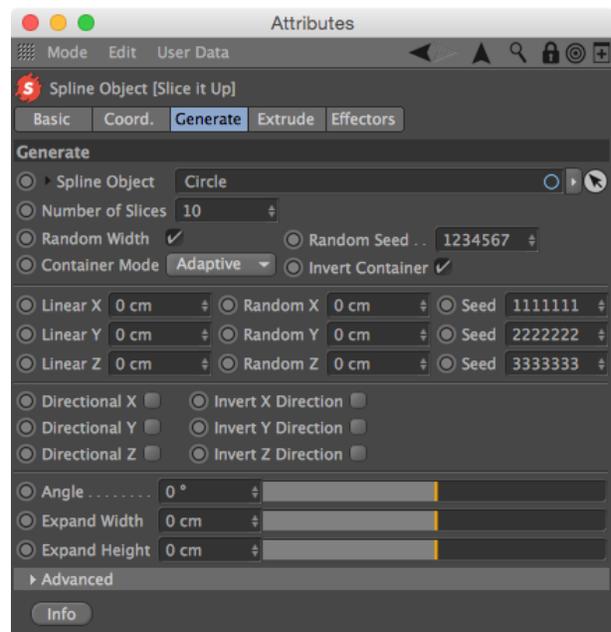
Generate tab

Under **Generate** tab you will find **Spline Object** link field where you drag and drop your spline of choice. This can be any kind of spline, as long as it's closed one.

Number of Slices - define how many slices you wish to have. You can go from 2 to infinity.

Random Width - enable this checkbox to randomise the width of each slice, as they are equal width by default.

Random Seed - generate a seed for the randomness of slice width. This option is available only if **Random Width** checkbox is enabled.



Container Mode - think of it as a geometric representation of a single full slice (or bounding box). There are three possible options: **Adaptive**, **Square** and **Circle**. Your choice must be made depending on which way and how you will be slicing and animating the object. Each of these options have its pros and cons, so please refer to the examples at the end of the documentation.

Invert Container - instead of slicing your spline object, plugin will slices it's container. In this case your spline object becomes as a mask object, that cuts through the container. Enable this checkbox to familiarise yourself how **Container Mode** works in different modes.

Use **Linear X** / **Linear Y** / **Linear Z** input fields to move each slice in a linear way. Every other slice will move in positive direction, while others will move in negative direction.

Random X / **Random Y** / **Random Z** introduces some level of randomness to the movement of each slice. Adjust the **Seed** to get different seed for the movement.

If slices get too far from each other using randomised way and there isn't much of randomness seen, try lowering down linear values to compensate the movement. Tweak both of these values to get desired look.

Slice movement is bi-directional by default, that is first moves up, second moves down, third up and so on. Enabling **Directional X / Y / Z** will force all slices to move in same direction.

Invert X / Y / Z Direction will invert movement values. This option is useful when you have all your keyframes set-up and want to change the overall direction of the movement.

Angle slider sets the rotation of the slice.

Expansion slider shrinks or grows the bounds of each slice. Pushing slider to the min will make object disappear and pushing it to the max will result in extraordinary transition of the object.

Expand Height works in similar way, but acts along Y axis. You should temporarily enable **Invert Container** checkbox to see how containers size is changing in the viewport.

Mask Width / Height Threshold - if masks width reaches this value (by lowering down the **Expand Width** slider) it will automatically be disabled. Adjust this value if you get errors when having really thin slice width.

Container Width / Height - containers size should not be equal to your splines bounding box, as errors are introduced when containers mask is intersecting slices mask. Enable **Invert Container** and change **Expand Width / Height** sliders to get better understanding about his control.

Extrude tab

Generate Mesh - enable this checkbox to generate geometry. This works in similar way as if you had **Slice it Up** object placed inside **Extrude Nurbs** object.

Other controls (Movement, Subdivision, Iso Subdivision, etc) are native Extrude Nurbs controls and will not be discussed in this documentation. Please refer to applications help files for more information about it.

Effectors tab

Fracture Mode is similar to Mograph's Fracture object: you have **Straight**, **Explode Segments** and **Explode Segments & Connect** features. Each of them works independently.

Straight mode will treat entire Slice it Up object as one piece, so effectors will influence it as the whole object and not its individual slices.

Explode Segments will treat each polygon of Slice it Up object as a separate clone. This way effectors will effect each and every polygon separately.

Explode Segments & Connect - a subsequent optimisation is carried out after the segments have been exploded that fuses overlapping vertices. In this case effectors will treat each slice as a separate clone. This mode is perfect if you are using effectors with falloff. However, make sure to add a Phong tag, as you will get some weird shading errors.

Effector - place all Effectors into this field that should affect the Slice it Up object. Arranging the Effectors in different sequences (top to bottom) in the Effectors field will lead to different results.

Container Mode

Adaptive

Pros - Slices cover entire shape; works perfect on any angle; no clipping.
Use this one if you are unsure;
Cons - Slice width will differ depending on the angle;
Container size is automatically recalculated depending on the angle;

Square

Pros - Slices cover entire shape; width does not change when changing angle; no clipping.
Cons - Some slices will be empty at certain angles, though rendering unnecessary geometry;
Container for the slices is calculated by the hypotenuse of your layer (the square the length of the hypotenuse equals the sum of the squares of the lengths of the width and hight of your object. *Wikipedia*)

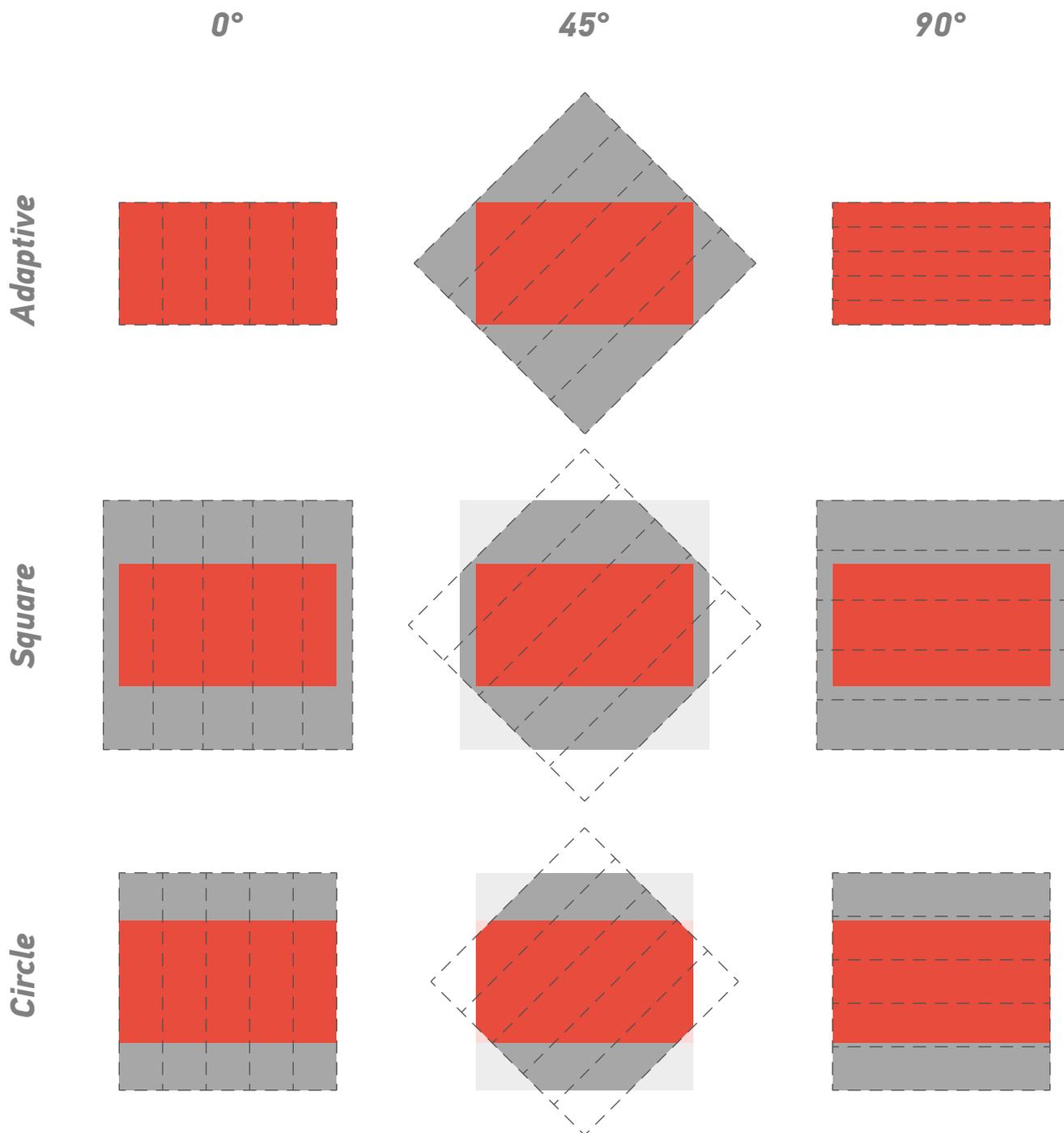
Circle

Pros - Perfect choice if your object is perfect circle.
Cons - Some clipping will appear if your object is not a perfect circle.
Container size is calculated by the largest side of your spline.

Following examples demonstrate the use of Square/Circle/Rectangle mode. In each case three different shapes (16x9 frame, perfect circle, and custom shape) are sliced with different mode. Please observe what

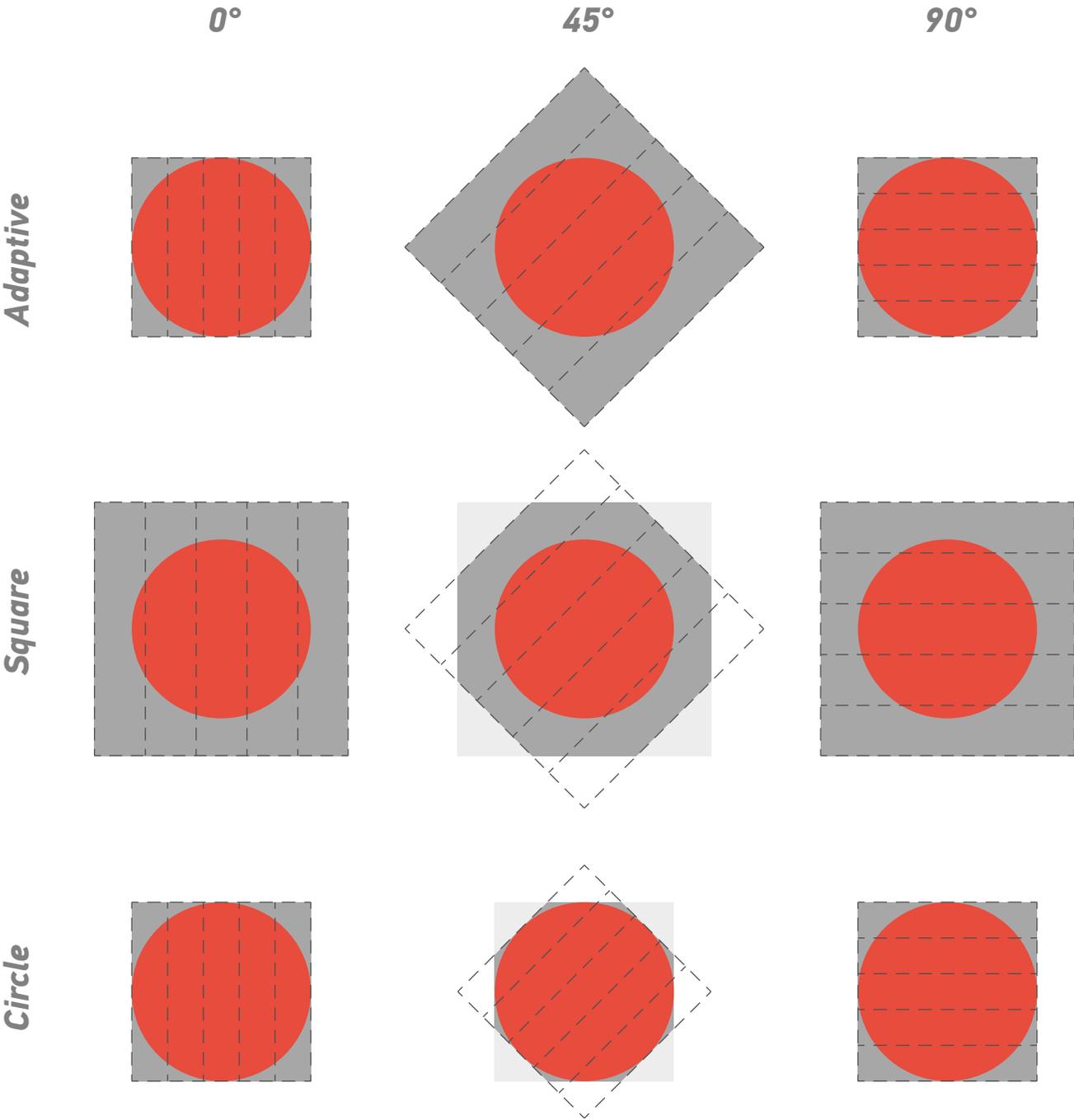
Shape - 16x9 frame.

Slice width differs on any angle in *Adaptive* mode. No clipping in *Square* mode, but empty slices at 90° angle. *Circle* mode - some clipping appear on the edges if angle is not 0°, 90°, 180°, 270°.



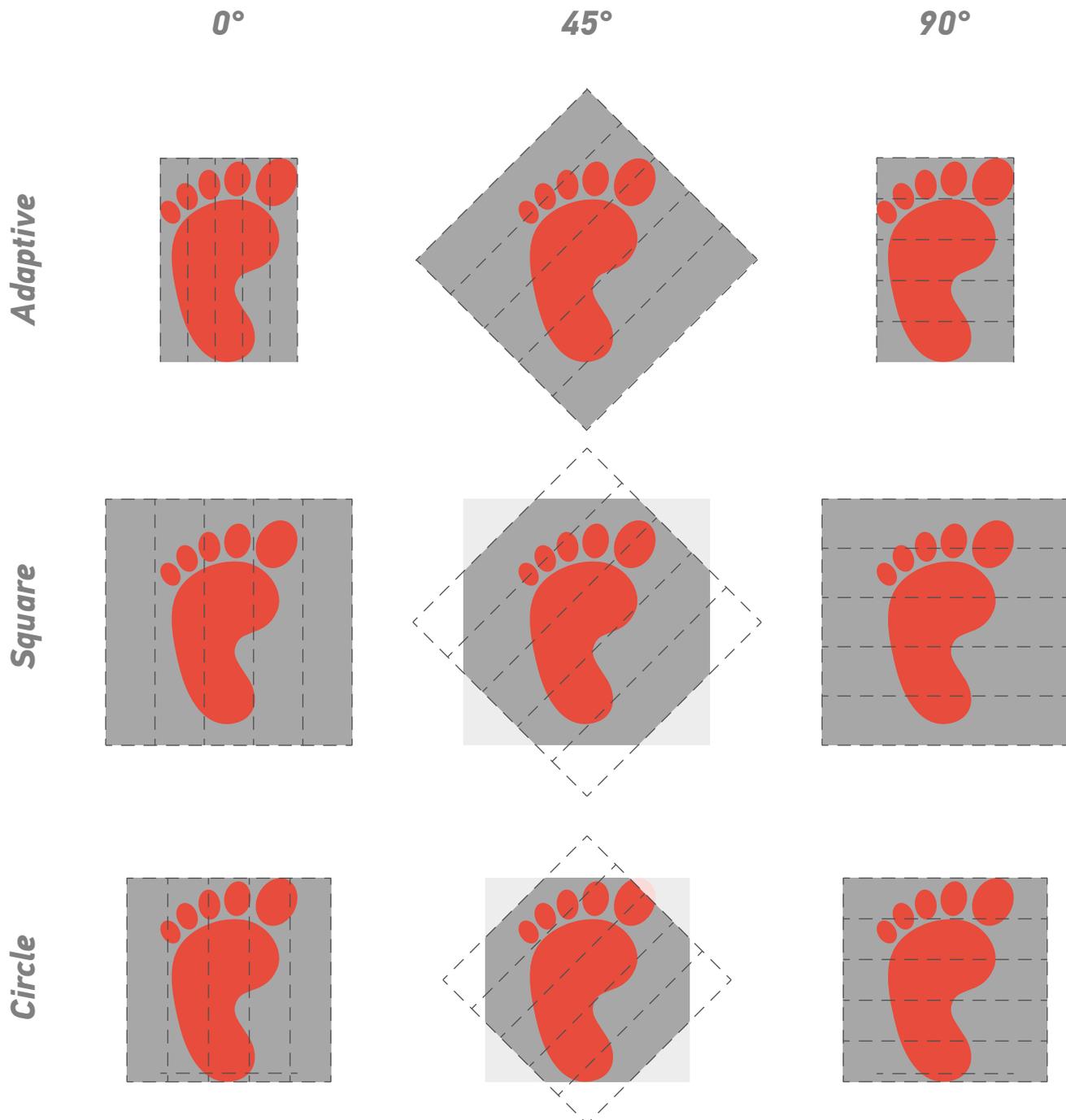
Shape - perfect circle.

No clipping and no empty slices appear at any angle in any mode. Perfect **Container mode** for this shape is *Circle*. Slices are evenly distributed along the object.



Shape - custom.

Probably best choice - *Adaptive* mode. No empty slices and no clipping. In this scenario empty slices in *Square* mode at 0° and small clipping at 45° in *Circle* mode.





Slice it Up

by Tomas Šinkūnas
www.rendertom.com

